



Welcome United States Patent and Trademark Office

[Search Session History](#)[BROWSE](#)[SEARCH](#)[IEEE XPLORE GUIDE](#)

Wed, 7 Mar 2007, 1:55:36 PM EST

Edit an existing query or
compose a new query in the
Search Query Display.

Search Query Display

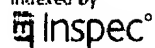
Select a search number (#)
to:

- Add a query to the Search Query Display
- Combine search queries using AND, OR, or NOT
- Delete a search
- Run a search

Recent Search Queries

- #1 ((cache<in>metadata) <and> (scheduler<in>metadata)
 <and> (fixed<in>metadata)
- #2 ((subsystem<in>metadata) <and> (scheduler<in>metadata))
 <and> (associated write<in>metadata)
- #3 ((cache<in>metadata) <and> (subsystem<in>metadata))
 <and> (schedules<in>metadata)
- #4 ((schedule fixed<in>metadata) <and> (read<in>metadata))
 <and> (write<in>metadata)
- #5 ((fixed number<in>metadata) <and> (read<in>metadata))
 <and> (write<in>metadata)
- #6 ((cache<in>metadata) <and> (schedule<in>metadata))
 <and> (reads and writes<in>metadata)
- #7 ((fixed<in>metadata) <and> (latency<in>metadata))<and>
 (pipeline<in>metadata)
- #8 ((fixed latency<in>metadata) <and> (pipeline<in>metadata))
 <and> (cache<in>metadata)

Indexed by

[Help](#) [Contact Us](#) [Privacy & :](#)

© Copyright 2006 IEEE -



USPTO

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide



THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used

cache scheduler associated write pipeline fixed predetermined latency

Found 570 of 198,310

Sort results by

☒ Save results to a Binder

[Try an Advanced Search](#)
[Try this search in The ACM Guide](#)

Display results

☒ Search Tips

☐ Open results in a new window

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐

1 [Query evaluation techniques for large databases](#)



Goetz Graefe

June 1993 **ACM Computing Surveys (CSUR)**, Volume 25 Issue 2

Publisher: ACM Press

Full text available: pdf(9.37 MB)

 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Database management systems will continue to manage large data volumes. Thus, efficient algorithms for accessing and manipulating large sets and sequences will be required to provide acceptable performance. The advent of object-oriented and extensible database systems will not solve this problem. On the contrary, modern data models exacerbate the problem: In order to manipulate large sets of complex objects as efficiently as today's database systems manipulate simple records, query-processi ...

Keywords: complex query evaluation plans, dynamic query evaluation plans, extensible database systems, iterators, object-oriented database systems, operator model of parallelization, parallel algorithms, relational database systems, set-matching algorithms, sort-hash duality

2 [VSV: L2-Miss-Driven Variable Supply-Voltage Scaling for Low Power](#)

Hai Li, Chen-Yong Cher, T. N. Vijaykumar, Kaushik Roy

 December 2003 **Proceedings of the 36th annual IEEE/ACM International Symposium on Microarchitecture MICRO 36**

Publisher: IEEE Computer Society

Full text available: pdf(205.58 KB)

 Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

Energy-efficient processor design is becoming more and more important with technology scaling and with high performance requirements. Supply-voltage scaling is an efficient way to reduce energy by lowering the operating voltage and the clock frequency of processors simultaneously. We propose a variable supply-voltage scaling (VSV) technique based on the following key observation: upon an L2 miss, the pipeline performs some independent computations but almost always ends up stalling and waiting for data, d ...

3 [Measuring Experimental Error in Microprocessor Simulation](#)

Rajagopalan Desikan, Doug Burger, Stephen W. Keckler



June 2001 **Proceedings of the 28th annual international symposium on Computer architecture ISCA '01**

Publisher: ACM Press

Full text available: pdf(237.69 KB) Additional Information: [full citation](#), [abstract](#), [citings](#), [index terms](#)

Abstract: We measure the experimental error that arises from the use of non-validated simulators in computer architecture research, with the goal of increasing the rigor of simulation- based studies. We describe the methodology that we used to validate a microprocessor simulator against a Compaq DS-10L workstation, which contains an Alpha 21264 processor. Our evaluation suite consists of a set of 21 microbenchmarks that stress different aspects of the 21264 microarchitecture. Using the microbenc ...

4 Experience Using Multiprocessor Systems—A Status Report



Anita K. Jones, Peter Schwarz

June 1980 **ACM Computing Surveys (CSUR)**, Volume 12 Issue 2

Publisher: ACM Press

Full text available: pdf(4.48 MB) Additional Information: [full citation](#), [references](#), [citings](#), [index terms](#)

5 Waiting algorithms for synchronization in large-scale multiprocessors



Beng-Hong Lim, Anant Agarwal

August 1993 **ACM Transactions on Computer Systems (TOCS)**, Volume 11 Issue 3

Publisher: ACM Press

Full text available: pdf(2.72 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citings](#), [index terms](#)

Through analysis and experiments, this paper investigates two-phase waiting algorithms to minimize the cost of waiting for synchronization in large-scale multiprocessors. In a two-phase algorithm, a thread first waits by polling a synchronization variable. If the cost of polling reaches a limit L_{poll} and further waiting is necessary, the thread is blocked, incurring an additional fixed cost, B . The choice of L_{poll}

Keywords: barriers, blocking, competitive analysis, locks, producer-consumer synchronization, spinning, waiting time

6 The KSR1: experimentation and modeling of poststore



E. Rosti, E. Smirni, T. D. Wagner, A. W. Apon, L. W. Dowdy

June 1993 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1993 ACM SIGMETRICS conference on Measurement and modeling of computer systems SIGMETRICS '93**, Volume 21 Issue 1.

Publisher: ACM Press

Full text available: pdf(1.28 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citings](#), [index terms](#)

Kendall Square Research introduced the KSR1 system in 1991. The architecture is based on a ring of rings of 64-bit microprocessors. It is a distributed, shared memory system and is scalable. The memory structure is unique and is the key to understanding the system. Different levels of caching eliminates physical memory addressing and leads to the ALLCACHE[®] scheme. Since requested data may be found in any of several caches, the initial access time is variable. Once pulled into the local ...


7 A parallel embedded-processor architecture for ATM reassembly

Richard F. Hobson, P. S. Wong

February 1999 **IEEE/ACM Transactions on Networking (TON)**, Volume 7 Issue 1

Publisher: IEEE Press

Full text available: Additional Information:

 [pdf\(331.21 KB\)](#)

[full citation](#), [references](#), [citations](#), [index terms](#)

Keywords: ATM, embedded systems, medium access control, segmentation and reassembly


8 [Trace-driven memory simulation: a survey](#)



Richard A. Uhlig, Trevor N. Mudge

June 1997 **ACM Computing Surveys (CSUR)**, Volume 29 Issue 2

Publisher: ACM Press

Full text available:  [pdf\(636.11 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

As the gap between processor and memory speeds continues to widen, methods for evaluating memory system designs before they are implemented in hardware are becoming increasingly important. One such method, trace-driven memory simulation, has been the subject of intense interest among researchers and has, as a result, enjoyed rapid development and substantial improvements during the past decade. This article surveys and analyzes these developments by establishing criteria for evaluating trac ...



Keywords: TLBs, caches, memory management, memory simulation, trace-driven simulation

9 [Multithreading II: Microarchitectural denial of service: insuring microarchitectural fairness](#)

Dirk Grunwald, Soraya Ghiasi

November 2002 **Proceedings of the 35th annual ACM/IEEE international symposium on Microarchitecture MICRO 35**

Publisher: IEEE Computer Society Press

Full text available:  [pdf\(996.00 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)
 [Publisher Site](#)

Simultaneous multithreading seeks to improve the aggregate computation bandwidth of a processor core by sharing resources such as functional units, caches, TLB and so on. To date, most research investigating the scheduling of these shared resources has focused on enhancing computational bandwidth. In this paper, we examine *scheduling fairness*. First, we show that a thread running on an implementation of a SMT processor can suffer from "denial of service" by a malicious thread, slowing dow ...


10 [Soft timers: efficient microsecond software timer support for network processing](#)



Mohit Aron, Peter Druschel

August 2000 **ACM Transactions on Computer Systems (TOCS)**, Volume 18 Issue 3

Publisher: ACM Press

Full text available:  [pdf\(272.44 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

This paper proposes and evaluates soft timers, a new operating system facility that allows the efficient scheduling of software events at a granularity down to tens of microseconds. Soft timers can be used to avoid interrupts and reduce context switches associated with network processing, without sacrificing low communication delays. More specifically, soft timers enable transport protocols like TCP to efficiently perform rate-based clocking of packet transmissions. Experiments indicate that ...

Keywords: polling, timers, transmission scheduling

11 Architectures and performance analysis: Scratchpad memory management for portable systems with a memory management unit



Bernhard Egger, Jaejin Lee, Heonshik Shin

October 2006 **Proceedings of the 6th ACM & IEEE International conference on Embedded software EMSOFT '06**

Publisher: ACM Press

Full text available: [pdf\(289.67 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

In this paper, we present a dynamic scratchpad memory allocation strategy targeting a horizontally partitioned memory subsystem for contemporary embedded processors. The memory subsystem is equipped with a memory management unit (MMU), and physically addressed scratchpad memory (SPM) is mapped into the virtual address space. A small minicache is added to further reduce energy consumption and improve performance. Using the MMU's page fault exception mechanism, we track page accesses and copy frequen ...

Keywords: code placement, compilers, heterogeneous memory, paging, portable systems, postpass optimization, scratchpad, virtual memory

12 Continual flow pipelines



Srikanth T. Srinivasan, Ravi Rajwar, Haitham Akkary, Amit Gandhi, Mike Upton

October 2004 **ACM SIGOPS Operating Systems Review , ACM SIGPLAN Notices , ACM SIGARCH Computer Architecture News , Proceedings of the 11th international conference on Architectural support for programming languages and operating systems ASPLOS-XI**, Volume 38 , 39 , 32 Issue 5 , 11 , 5

Publisher: ACM Press

Full text available: [pdf\(274.26 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Increased integration in the form of multiple processor cores on a single die, relatively constant die sizes, shrinking power envelopes, and emerging applications create a new challenge for processor architects. How to build a processor that provides high single-thread performance and enables multiple of these to be placed on the same die for high throughput while dynamically adapting for future applications? Conventional approaches for high single-thread performance rely on large and complex co ...

Keywords: CFP, instruction window, latency tolerance, non-blocking

13 Enhancing software reliability with speculative threads



Jeffrey Oplinger, Monica S. Lam

October 2002 **ACM SIGPLAN Notices , ACM SIGOPS Operating Systems Review , ACM SIGARCH Computer Architecture News , Proceedings of the 10th international conference on Architectural support for programming languages and operating systems ASPLOS-X**, Volume 37 , 36 , 30 Issue 10 , 5 , 5

Publisher: ACM Press

Full text available: [pdf\(1.47 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

This paper advocates the use of a monitor-and-recover programming paradigm to enhance the reliability of software, and proposes an architectural design that allows software and hardware to cooperate in making this paradigm more efficient and easier to program. We propose that programmers write monitoring functions assuming simple sequential execution semantics. Our architecture speeds up the computation by executing the monitoring functions speculatively in parallel with the main computation. For ...

14 Scalable Store-Load Forwarding via Store Queue Index Prediction


Tingting Sha, Milo M. K. Martin, Amir Roth

November 2005 **Proceedings of the 38th annual IEEE/ACM International Symposium on Microarchitecture MICRO 38**

Publisher: IEEE Computer Society

Full text available:  [pdf\(306.61 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [index terms](#)

 [Publisher Site](#)


Conventional processors use a fully-associative store queue (SQ) to implement store-load forwarding. Associative search latency does not scale well to capacities and bandwidths required by wide-issue, large window processors. In this work, we improve SQ scalability by implementing store-load forwarding using speculative indexed access rather than associative search. Our design uses prediction to identify the single SQ entry from which each dynamic load is most likely to forward. When a load exec ...

15 Technical papers: Imaging and visual analysis---Large image correction and warping in a cluster environment


Vijay S. Kumar, Benjamin Rutt, Tahsin Kurc, Umit Catalyurek, Joel Saltz, Sunny Chow, Stephan Lamont, Maryann Martone

November 2006 **Proceedings of the 2006 ACM/IEEE conference on Supercomputing SC '06**

Publisher: ACM Press

Full text available:  [pdf\(394.33 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#)

 [html\(1.86 KB\)](#)

This paper is concerned with efficient execution of a pipeline of data processing operations on very large images obtained from confocal microscopy instruments. We describe parallel, out-of-core algorithms for each operation in this pipeline. One of the challenging steps in the pipeline is the warping operation using inverse mapping based methods. We propose and investigate a set of algorithms to handle the warping computations on storage clusters. Our experimental results show that the proposed ...


Keywords: PC clusters, digital microscopy, imaging, out-of-core, parallel computation, warping

16 Information and control in gray-box systems

Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau

October 2001 **ACM SIGOPS Operating Systems Review , Proceedings of the eighteenth ACM symposium on Operating systems principles SOSP '01**, Volume 35 Issue 5

Publisher: ACM Press

Full text available:  [pdf\(1.59 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

In modern systems, developers are often unable to modify the underlying operating system. To build services in such an environment, we advocate the use of *gray-box* techniques. When treating the operating system as a gray-box, one recognizes that not changing the OS restricts, but does not completely obviate, both the *information* one can acquire about the internal state of the OS and the *control* one can impose on the OS. In this paper, we develop and investigate three gray-bo ...

17 Implementation and evaluation of a QoS-capable cluster-based IP router

Prashant Pradhan, Tzi-cker Chiueh

November 2002 **Proceedings of the 2002 ACM/IEEE conference on Supercomputing Supercomputing '02**

Publisher: IEEE Computer Society Press

Full text available:  pdf(215.68 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

A major challenge in Internet edge router design is to support both high packet forwarding performance and versatile and efficient packet processing capabilities. The thesis of this research project is that a cluster of PCs connected by a high speed system area network provides an effective hardware platform for building routers to be used at the edges of the Internet. This paper describes a scalable and extensible edge router architecture called *Panama*, which supports a novel aggregate r ...


18 Multicast Video-on-Demand services



Huadong Ma, Kang G. Shin

January 2002 **ACM SIGCOMM Computer Communication Review**, Volume 32 Issue 1

Publisher: ACM Press

Full text available:  pdf(1.28 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The server's storage I/O and network I/O bandwidths are the main bottleneck of VoD service. Multicast offers an efficient means of distributing a video program to multiple clients, thus greatly improving the VoD performance. However, there are many problems to overcome before development of multicast VoD systems. This paper critically evaluates and discusses the recent progress in developing multicast VoD systems. We first present the concept and architecture of multicast VoD, and then introduce ...

Keywords: Quality-of-Service (QoS), VCR-like interactivity, Video-on-Demand (VoD), multicast, scheduling

19 Real-time shading



Marc Olano, Kurt Akeley, John C. Hart, Wolfgang Heidrich, Michael McCool, Jason L. Mitchell, Randi Rost

August 2004 **ACM SIGGRAPH 2004 Course Notes SIGGRAPH '04**

Publisher: ACM Press

Full text available:  pdf(7.39 MB) Additional Information: [full citation](#), [abstract](#)

Real-time procedural shading was once seen as a distant dream. When the first version of this course was offered four years ago, real-time shading was possible, but only with one-of-a-kind hardware or by combining the effects of tens to hundreds of rendering passes. Today, almost every new computer comes with graphics hardware capable of interactively executing shaders of thousands to tens of thousands of instructions. This course has been redesigned to address today's real-time shading capabili ...


20 System-level power optimization: techniques and tools



Luca Benini, Giovanni de Micheli

April 2000 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 5 Issue 2

Publisher: ACM Press

Full text available:  pdf(385.22 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This tutorial surveys design methods for energy-efficient system-level design. We consider electronic systems consisting of a hardware platform and software layers. We consider the three major constituents of hardware that consume energy, namely computation, communication, and storage units, and we review methods of reducing their energy consumption. We also study models for analyzing the energy cost of software, and methods for energy-efficient software design and compilation. This survey ...

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2007 ACM, Inc.
[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)